A polynomial texture extraction with application in dynamic texture classification

Redouane El Moubtahij, Bertrand Augereau, Christine Fernandez-Maloigne, Hamid Tairi

A polynomial texture extraction with application in dynamic texture classification

R. EL MOUBTAHIJ[a,b], B. AUGEREAU[a], C. FERNANDEZ-MALOIGNE[a] and H. TAIRI[b]

[a]XLIM-SIC Laboratory, UMR CNRS 7252 Bd Marie et Pierre Curie, 86962 Chasseneuil France
e-mail: {redouane.el.moubtahij, bertrand.augereau, christine.fernandez}@univ-poitiers.fr
[b]LIIAN laboratory - Faculty of Science Dhar EL Mahraz - USMBA,30003 Fez Morocco
e-mail: htairi@yahoo.fr

ABSTRACT

Geometry and texture image decomposition is an important paradigm in image processing. Following to Yves Meyer works based on Total Variation (VT), the decomposition model has known a renewed interest. In this paper, we propose an algorithm which decomposes color image into geometry and texture component by projecting the image in a bivariate polynomial basis and considering the geometry component as the partial reconstruction and the texture component as the remaining part. The experimental results show the adequacy of using our method as a texture extraction tool. Furthermore, we integrate it into a dynamic texture classification process.

Keywords: Texture extraction, Dynamic Textures, Polynomial decomposition, Video classification.

1. INTRODUCTION

Various works, more or less recently, have tackled the problem of image decomposition that still is one of the major aims in image processing. In fact, a lot of domain are concerned, from denoising to pattern recognition. In this paper, we investigate the possibility of having a representation space of the image information, space adapted to video classification and more especially dynamic textures classification. After a brief reminding of the principle of image decomposition, we detail our method of texture extraction : firstly presenting the Polynomial Transform, then the decomposition step. After that, our decomposition method is used to construct a video descriptor for dynamic texture classification.

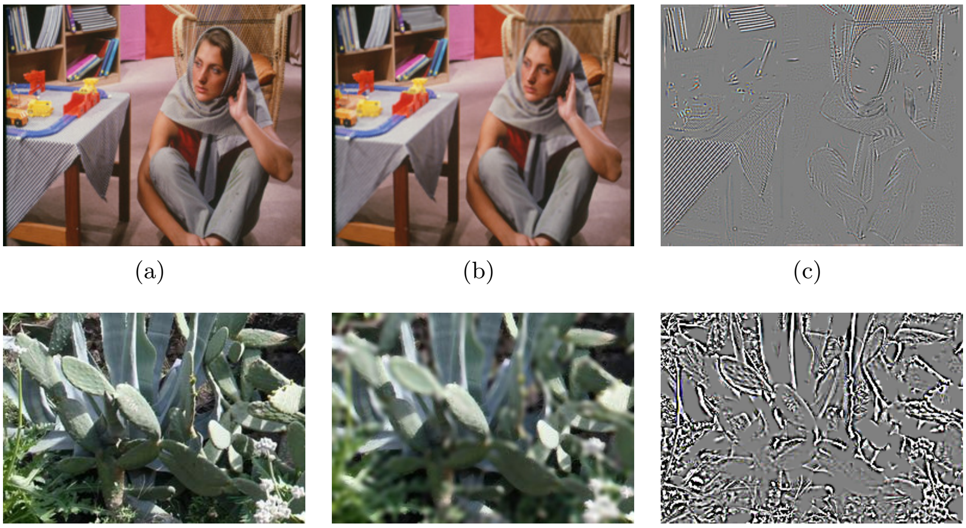
2. IMAGE DECOMPOSITION

Yves Meyer[1] has proposed a model of image decomposition using the algorithm of Rudin-Osher-Fatemi.[2] According to this model, an image is split in two parts, one containing the structure u, the other one containing the texture u. The result is provided by the minimization of the functional

$$\mathcal{F}(u,v) = \|f\|_F + \lambda \|g\|_G \tag{1}$$

where $f \in F$, $g \in G$ and λ is the parameter of the model. More precisely, F is the space of functions with bounded variations and G the space of oscillating functions with the property that more a function is oscillating, more its standard norm $\|g\|_G$ will be low. This model can be solved numerically due to the formulation proposed in J-F.Aujol,[3,4] by the introduction of an additional parameter μ corresponding to the maximum norm of textures in the space G. The use of non-linear projectors defined by A.Chambolle[5] provides the decomposition of the image by an iterative algorithm (see[3,4] for more details).

A. Buades[6] has created a method that, as we know, is the fastest and most efficient implementation of the theory given Yves Meyer.[1] It is a fast approximate solution to the original variational problem obtained by applying non-linear filtering to the image. For each image pixel, a decision is made whether it belongs to the geometric part or to the texture part. This decision is made by computing a local total variation of the image around the point, and comparing it to the local total variation after a low pass filter has been applied. In fact, edge points in an image tend to have a slowly varying local total variation when the image is convoluted by a low pass filter while textural points instead show a strong decay of their local total variation. After the selection of the points belonging to the geometrical part, the texture part is considered as the difference between the original image and the geometrical part. (See Figure 1 for image decomposition with Buades[6] method). In fact, there is no unique decomposition and the algorithm relies on an important parameter, the scale parameter which is directly related to the granularity of textures distinguished.



(a) (b) (c)

(d) (e) (f)

Figure 1. A. Buades method with scale parameter set to 3 : (a and d) original images, (b and e) images of geometry, (c and f) images of texture.

3. IMAGE DECOMPOSITION WITH POLYNOMIAL TRANSFORM

A *Bivariate Polynomial (BP)* of degree d is a function of $x = (x_1, x_2) \in \mathbb{R}^2$ given by

$$P(x) = \sum_{\substack{(d_1,d_2)\in[0;d]^2 \\ d_1+d_2\leq d}} a_{d_1,d_2}\, x_1^{d_1}\, x_2^{d_2} \tag{2}$$

with any $a_{d_1,d_2} \in \mathbb{R}$.

3.1 Bivariate polynomial basis

Considering a finite set of pairs $D = \{(d_1, d_2)\} \subset \mathbb{N}^2$, we represent by $\mathbb{E}_D$ the space of all BP such as $a_{d_1,d_2} \equiv 0$ if $(d_1, d_2) \notin D$ and by $\mathcal{K}_D$ the subset of monomials

$$\mathcal{K}_D = \left\{K_{d_1,d_2}(x) = x_1^{d_1}\, x_2^{d_2}\right\}_{(d_1,d_2)\in D} \tag{3}$$

Obviously $\mathcal{K}_D$ satisfies the linear independence and spanning conditions and so, $\mathcal{K}_D$ is a basis of $\mathbb{E}_D$, the canonical basis. In our context of color image decomposition, we look for bases with more suitable properties such as orthogonality or normality. So, to construct a *discrete orthonormal BP finite basis* we first have to consider the underlying discrete domain

$$\Omega = \left\{x_{(u,v)} = \left(x_{1,(u,v)}, x_{2,(u,v)}\right)\right\}_{(u,v)\in D} \tag{4}$$

where D will now represent the set of pairs associated to Ω. Starting from $\mathcal{K}_D$ we intend to construct a new orthonormal basis applying the Gram-Schmidt process. That implies that we need some product and norm for functions defined on Ω. Given two bivariate functions, F and G, their *discrete extended scalar product* is defined by

$$\langle F|G\rangle = \sum_{(u,v)\in D} \omega(x_{(u,v)})\, F(x_{(u,v)})\, G(x_{(u,v)}) \tag{5}$$

with ω a real positive function over Ω [Legendre, Chebichev, Hermite, ...]. Then, the actual construction process of an orthonormal basis

$$\mathcal{B}_{D,\omega} = \{B_{d_1,d_2}\}_{(d_1,d_2)\in D} \tag{6}$$

is a recurrence upon (d_1, d_2)

$$T_{d_1,d_2}(x) = K_{d_1,d_2}(x) - \sum_{(l_1,l_2)\prec_2(d_1,d_2)} \langle K_{d_1,d_2}|B_{l_1,l_2}\rangle_\omega B_{l_1,l_2}(x) \tag{7}$$

$$B_{d_1,d_2}(x) = \frac{T_{d_1,d_2}(x)}{|T_{d_1,d_2}|_\omega} \tag{8}$$

where $\prec_2$ is the lexicographical order and $|\ |_\omega$ the norm induced by $\langle\ |\ \rangle_\omega$. The resulting set of B polynomials verifies

$$\langle B_{d_1,d_2}|B_{l_1,l_2}\rangle_\omega = \begin{cases} 0 & \text{if } (d_1,d_2) \neq (l_1,l_2) \\ 1 & \text{if } (d_1,d_2) = (l_1,l_2) \end{cases} \tag{9}$$

and so $\mathcal{B}_{D,\omega}$ is effectively an orthonormal basis with respect to a weighting function ω. A special case, later used in this paper, is the *complete base* where D exactly represents the set of pairs associated to Ω, that is

$$D = [0; N_1] \times [0; N_2] \tag{10}$$

The space $\mathbb{E}_D$ being dense in the space of functions over Ω, it allows to well approximate any bivariate function I by an appropriate combination of elements of a $\mathcal{B}_{D,\omega}$ orthonormal basis

$$P_I(x) = \sum_{\{(d_1,d_2)\}\subset D} b_{d_1,d_2}\, B_{d_1,d_2}(x) \tag{11}$$

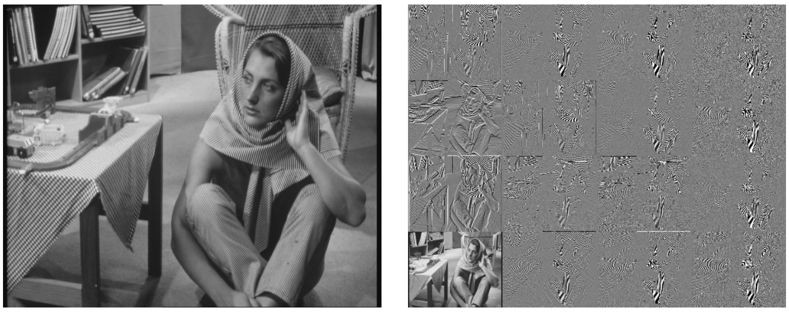
where b_{d_1,d_2} is the scalar resulting of the projection $b_{d_1,d_2} = \langle I|B_{d_1,d_2}\rangle_\omega$. In fact, with a complete orthonormal basis, the polynomial approximation of I is a first order osculatory polynomial interpolation : for all points of the domain we have $P_I(x) = I(x)$. An other nice property is that the projection on polynomial B_{d_1,d_2} can be considered as an approximation of the partial derivation $\partial_{d_1}\partial_{d_2}$. Finally and in practice, the discrete projection process supposes that both I and B_{d_1,d_2} can be evaluated on the common domain Ω. So, the set of collocation points can be obtain by uniform or non-uniform discretization of given intervals. For example, with $[-1;1]^2$ and referring to equation (10), the collocation points obtained by uniform discretization are

$$x_{1,(u,v)} = -1 + \frac{2u}{N_1} \qquad x_{2,(u,v)} = -1 + \frac{2v}{N_2} \tag{12}$$

3.2 Polynomial Transform

Now we describe the Polynomial Transform algorithm which is founded on piecewise discrete polynomial approximation and the principle of Wavelet Packet. At a given level of this multi-resolution transform, lets consider a function U defined on a domain Ω of size $n_1 \times n_2$, and a basis $\mathcal{B}_{M,\omega}$ defined on a support M of size $h_1 \times h_2$, the transform process is defined as follows :

1. definition of a covering set of the discrete domain Ω with sub-domains Ω_M of size $h_1 \times h_2$
2. for each sub-domain Ω_M, projection of the corresponding restriction U_M in the basis $\mathcal{B}_{M,\omega}$ that provides the coefficients $b_{M,d_1,d_2} = \langle U_M|B_{d_1,d_2}\rangle_\omega$



(a) (b)

Figure 2. First level of Polynomial Transform with a 4×4 Hermite complete basis.

3. for all pair (d_1, d_2) the reordering of the global set of coefficients b_{M,d_1,d_2} into $h_1 \times h_2$ new functions U_{d_1,d_2} defined on domains of size $n_1|_{h_1} \times n_2|_{h_2}$

This method provides some flexibility especially in the choice of the resolution factors, depending of the sub-domains size and of their offset, the transform can be perform with juxtaposed or overlapped sub-domains. Moreover, the choice of the weighting function ω allows, at the same time, to perform a multi-scale and a multi-resolution transformation. As an illustration, Figure 2 shows an example of a first level transformation using a 4×4 Hermite complete basis.

3.3 Image decomposition by partial reconstruction

A Polynomial Transform performed with a complete basis is perfectly reversible. However, it is possible to obtain many kind of approximations by selecting the coefficients during the reconstruction phase, i.e. *partial reconstruction*. This choice of coefficients may follow various strategies, among them we have : (a) brutal restriction to a given subset, for example the polynomials of degree less than a threshold; (b) restriction based on energies, for example by using the normality of the basis to assimilate the absolute value of its coefficients to a part of the energy of a sub-domain, then sort the coefficients and finally retain a fixed number of these coefficients or those satisfying a certain condition (cf. PCA).

In our case, to decompose the image into geometric and texture component, we assume that the geometrical part is given by a partial reconstruction $\tilde{I}$ of the original image I in an overlapped Polynomial Transform context. As seen before, this transform is very flexible, so there are many conceivable solutions. In order to get a compromise between quality and computation time, we choose to use an Hermite basis, to set the sub-domain size to 3×3, with an offset of 2, and to select the three dominant coefficients for each sub-domain. The partial reconstruction of a color image $I = (I_j)_{j=1\cdots 3}$, i.e. the construction of the geometrical part, can then be summarize by

$$\tilde{I}_j(x) = \frac{1}{c(x)} \sum_{\{\Omega_M \ni x\}} \left(\Psi(\Omega_M)\, \omega(x_M) \sum_{(d_1,d_2)\in \mathcal{P}_{j,M}} b_{j,d_1,d_2}(I_{j,M})\, B_{d_1,d_2}(x_M) \right) \tag{13}$$

where x is a point referring to the global image domain Ω, x_M is the same point referring to a given sub-domain Ω_M, $I_{j,M}$ the restriction of I_j to sub-domain Ω_M, $\mathcal{P}_{j,M}$ the set of selected polynomials for $I_{j,M}$ approximation, ω the weighting function of the scalar product and b_{j,d_1,d_2} the coefficient of the projection of $I_{j,M}$ on the basis polynomial B_{d_1,d_2}. A degree of anisotropy $\Psi(\Omega_M)$ is assigned to each sub-domain Ω_M and $c(x)$ is the sum of x contributions, $c(x) = \sum_{\{\Omega_M \ni x\}} \Psi(\Omega_M)\, \omega(x_M)$. The degree of anisotropy is evaluated according to

$$\Psi(\Omega_M) = \frac{1}{1 + \lambda^r} \tag{14}$$

where λ is the largest eigenvalue of a color structure tensor composed with the approximations of partial derivatives, projections on the basis polynomials of degree one

$$\mathcal{S} = \begin{pmatrix} \sum_j (b_{j,1,0})^2 & \sum_j b_{j,1,0}\, b_{j,0,1} \\ \sum_j b_{j,0,1}\, b_{j,1,0} & \sum_j (b_{j,0,1})^2 \end{pmatrix} \tag{15}$$

The balance between isotropic and anisotropic reconstruction is adjust by the parameter r that controls the degree of anisotropy in a range of 0.25 for isotropic (gaussian) to 2 for highly isotropic. By doing that, we assure a real color process and avoid marginal treatment deficiencies. Finally, the texture component I^T is simply deduce from the partial reconstruction by considering that it is the residual part of the image

$$I^T = I - \tilde{I} \tag{16}$$

The results of image decomposition into geometry and texture components by partial reconstruction after an

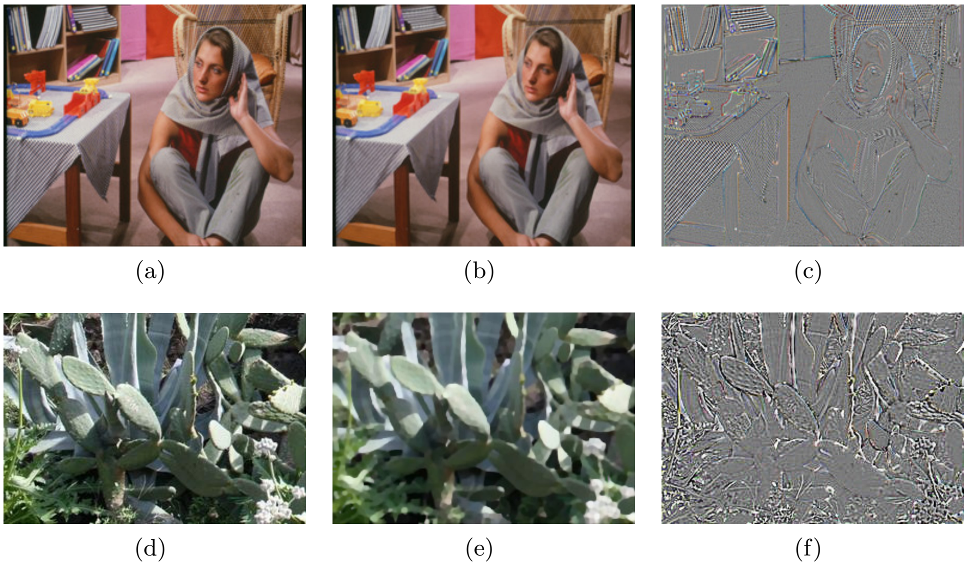


Figure 3. Our decomposition method with $r = 0.75$: (a and d) original images, (b and e) images of geometry, (c and f) images of texture.

Hermite Polynomial Transform as defined in equation(13), with the parameter r of equation (14) set to 0.75, are shown in Figure 3.

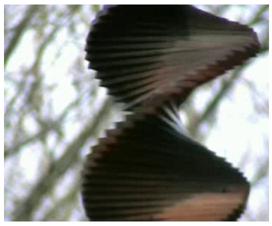
4. EXPERIMENTAL RESULTS

To illustrate the efficiency of our method, we will use it in a dynamic texture classification scheme which is among challenging research themes in the image analysis field. The database used in our application is Dyntex++ database,[7] see Figure 4. Because it is composed of images that represent textures in motion, it therefore seems appropriate for the present study. Since the videos are not evenly distributed among the 36 classes in this database, the pretreatments[7] were realized so that each class contains the same number of sequences. We finally have 3600 dynamic textures, grouped into 36 classes and each sequence, i.e. each instance of a dynamic texture, has a size of $50 \times 50 \times 50$ pixels.

In our approach, we use the video sequences without converting them in grey level. Then, we both extract the temporal information provided by the color displacement fields[14] (ν_1, ν_2) and the spatial information provided by the color texture components I^T using our method of Polynomial Transform decomposition. Thereafter, we modelize ν_1, ν_2 and I^T by approximating them with a complete Hermite bivariate basis of degree 2. In our experiment the datas were randomly split into two equal size training and test sets.[9–13] The random split was

(a) Blossoming

(b) Flag



(c) Rotating wind ornament

(d) Underwater

(e) Vehicle traffic on road

(f) Rain on water

Figure 4. Examples of Dyntex++ database.

VLBP (Volume Local Binary Patterns)[8]	61.1%
SIFT-3D (3 Dimensional Scale Invariant Feature Transform)[9]	63.7%
LBP-TOP(Local Binary Patterns on Three Orthogonal Planes)[10,11]	71.2%
DFS (Dynamic Fractal Spectrum)[12]	89,9%
HOG-NSP (Histogram of Oriented Gradients with Nine Symmetry Planes)[11]	90,1%
NLSSA (Non-Linear Stationary Subspace Analysis)[13]	92.4%
Our method	93,13%

Table 1. Classification results for Dyntex++ database.

repeated 10 times and the average classification accuracy is reported in Table 1. The classification is performed with a SVM[15] as classifier and RBF functions as kernels. From Table 1, we can see that our descriptor performs quite well comparatively to the usual methods of dynamic textures classification. Moreover, the application of our approach provides a very significant gain in computation time because all main computations can be realized through convolutions.

5. CONCLUSION

In this paper, we have proposed a new approach for texture extraction from color image sequence, by using Polynomial Transformations. Partial reconstruction and global approximation are used to build descriptors used in a classification process of dynamic textures. In addition to the simplicity of implementation, we provide a computing time which is especially fast compared to most of methods based on the theory of Yves Meyer due to the cost of the minimization of the total variation. The experimental results show that the proposed approach achieves a very good recognition rate for the Dyntex++ database. This shows the relevance of our texture extraction method in the context of classification of dynamic textures. In some future, we will continue to improve our image decomposition method in order to extract the noise coefficients ignored in the partial reconstruction of the image. We will also investigate the abilities of a derived method which only relies on three dimensional transformations for our classification process of dynamic textures.

REFERENCES

[1] Meyer, Y., [*Oscillating Patterns in Image Processing and Nonlinear Evolution Equations: The Fifteenth Dean Jacqueline B. Lewis Memorial Lectures*], Memoirs of the American Mathematical Society, American Mathematical Society (2001).

[2] Rudin, L. I., Osher, S., and Fatemi, E., "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena* **60**(14), 259 – 268 (1992).

[3] Jean-François, A., Aubert, G., Blanc-Féraud, L., and Chambolle, A., "Image decomposition: application to textured images and SAR images," Tech. Rep. RR-4704 (Jan. 2003).

[4] Jean-François, A., Gilboa, Guy, Chan, Tony, Osher, and Stanley, "Structure-texture image decomposition-modeling, algorithms, and parameter selection," *International Journal of Computer Vision* **67**(1), 111–136 (2006).

[5] Chambolle and Antonin, "An algorithm for total variation minimization and applications," *Journal of Mathematical Imaging and Vision* **20**(1-2), 89–97 (2004).

[6] Buades, A., Triet, L., Jean-Michel, M., and Luminita, V., "Cartoon/Texture Image Decomposition ," *Image Processing On Line* **1** (2011).

[7] Péteri, R., Fazekas, S., and Huiskes, M. J., "DynTex : a Comprehensive Database of Dynamic Textures," *Pattern Recognition Letters* **Volume 31, Issue 12**, 1627–1632 (Sept. 2010). http://projects.cwi.nl/dyntex/.

[8] Zhao, G. and Pietikinen, M., "Dynamic texture recognition using volume local binary patterns," in [*Dynamical Vision*], *Lecture Notes in Computer Science* **4358**, 165–177, Springer Berlin Heidelberg (2007).

[9] Scovanner, P., Ali, S., and Shah, M., "A 3-dimensional sift descriptor and its application to action recognition," in [*Proceedings of the 15th International Conference on Multimedia*], *MULTIMEDIA '07*, 357–360, ACM, New York, NY, USA (2007).

[10] Zhao, G. and Pietikainen, M., "Dynamic texture recognition using local binary patterns with an application to facial expressions," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **29**, 915–928 (June 2007).

[11] Ehsan, N., Harandi, M., Bigdeli, A., Baktash, M., Postula, A., and Lovell, B., "Directional space-time oriented gradients for 3d visual pattern analysis," in [*Computer Vision ECCV 2012*], **7574**, 736–749 (2012).

[12] Xu, Y., Quan, Y., Ling, H., and Ji, H., "Dynamic texture classification using dynamic fractal analysis," in [*Computer Vision (ICCV), 2011 IEEE International Conference on*], 1219–1226 (Nov 2011).

[13] Baktashmotlagh, M., Harandi, M., Lovell, B., and Salzmann, M., "Discriminative non-linear stationary subspace analysis for video classification," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **36**, 2353–2366 (Dec 2014).

[14] Augereau, B. and Fernandez-Maloigne, B. T. C., "Vectorial Computation of the Optical Flow in Color Image Sequences," in [*Thirteenth Color Imaging Conference*], **13**, 130–134 (Nov. 2005).

[15] Chih-Chung, C. and Chih-Jen, L., "Libsvm: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.* **2**, 27:1–27:27 (May 2011).